AMENDMENTS TO THE CLAIMS

1. (currently amended) A method for executing database transactions, comprising the steps of:

>defining each node of a plurality of interconnected nodes in terms of
>>processor and storage resources of a parallel computing system;

>mapping a first set of virtual processors across a first subset of the nodes to
>>create a first map with at least one virtual processor being mapped to
>>each node in the first subset;

>mapping a second set of virtual processors across a second subset of the
>>nodes to create a second map with at least one virtual processor
>>being mapped to each node in the second subset;

>storing the first map as a first configuration and the second map as a second
>>configuration; and

>executing at least one transaction using the first set of virtual processors
>>and simultaneously executing at least one transaction using the
>>second set of virtual processors;

>wherein executing a transaction comprises:

>>identifying tasks necessary to complete the transaction;

>>identifying storage resources necessary to complete each of the
>>>tasks;

>>defining one or more groups of virtual processors having access to
>>>the identified storage resources as one or more transaction
>>>groups;

>>assigning to each task one of the transaction groups that includes
>>>virtual processors having access to resources necessary to
>>>complete the task; and

2

completing each task using the virtual processors of the assigned

transaction group.

2. (original) The method of claim 1, wherein the first subset of the nodes includes all of the plurality of nodes and the second subset of nodes includes all of the plurality of nodes except for at least one failed node.

3. (original) The method of claim 2, further comprising the steps of:

mapping a third set of virtual processors across a third subset of the nodes to create a third map with at least one virtual processor being mapped to the third subset;

storing the third map as a third configuration; and wherein the third subset of the nodes includes all of the plurality of nodes.

4. (original) The method of claim 3, wherein the third subset includes at least one restored node.

5. (original) The method of claim 2, wherein at least one transaction using the first set of virtual processors does not use any virtual processors mapped to the at least one failed node.

6. (original) The method of claim 2, further comprising the step of:

halting an impacted transaction, the impacted transaction using the first set of virtual processors including at least one virtual processor mapped to one of the at least one failed nodes.

7. (original) The method of claim 1, wherein the first and second configurations are stored as a configuration table and configurations are removed from the table when they are no longer associated with an executing transaction.

8. (cancelled)

9. (currently amended) The method of <u>claim 1</u> ~~claim 8~~, wherein using the first set of virtual processors consists of completing tasks assigned to transaction groups that include virtual processors from the first set of virtual processors.

10. (original) The method of claim 1, wherein executing at least one transaction using the first set of virtual processors simultaneously with executing at least one transaction using the second set of virtual processors comprises executing a first transaction having at least first and second tasks, the first task executed by one or more virtual processors of the first set and the second task executed by one or more virtual processors of the second set.

11. (currently amended) A computer program, stored on a tangible storage medium, for use in executing database transactions, the program comprising executable instructions that cause a computer to

define each node of a plurality of interconnected nodes in terms of

processor and storage resources of a parallel computing system;

map a first set of virtual processors across a first subset of the nodes to

create a first map with at least one virtual processor being mapped to

each node in the first subset;

map a second set of virtual processors across a second subset of the nodes

to create a second map with at least one virtual processor being

mapped to each node in the second subset;

store the first map as a first configuration and the second map as a second

configuration; and

execute at least one transaction using the first set of virtual processors and

simultaneously execute at least one transaction using the second set

of virtual processors;

wherein executing a transaction comprises:

identifying tasks necessary to complete the transaction;

identifying storage resources necessary to complete each of the

tasks;

defining one or more groups of virtual processors having access to

the identified storage resources as one or more transaction

groups;

assigning to each task one of the transaction groups that includes

virtual processors having access to resources necessary to

complete the task; and

completing each task using the virtual processors of the assigned

transaction group.

12. (original) The computer program of claim 11, wherein the first subset of the nodes includes all of the plurality of nodes and the second subset of nodes includes all of the plurality of nodes except for at least one failed node.

13. (original) The computer program of claim 12, further comprising executable instructions that cause a computer to:

map a third set of virtual processors across a third subset of the nodes to

create a third map with at least one virtual processor being mapped

to each node in the third subset;

store the third map as a third configuration; and wherein the third subset of

the nodes includes all of the plurality of nodes.

14. (original) The computer program of claim 13, wherein the third subset includes at least one restored node.

15. (original) The computer program of claim 12, wherein the at least one transaction using the first set of virtual processors does not use any virtual processors mapped to the at least one failed node.

16. (original) The computer program of claim 12, further comprising executable instructions that cause a computer to:

      halt an impacted transaction, the impacted transaction using the first set of
            virtual processors including at least one virtual processor mapped to
            one of the at least one failed nodes.

17. (original) The computer program of claim 11, wherein the first and second configurations are stored as a configuration table and configurations are removed from the table when they are no longer associated with an executing transaction.

18. (cancelled)

19. (currently amended) The computer program of <u>claim 11</u> ~~claim 18~~, wherein using the first set of virtual processors consists of completing tasks assigned to transaction groups that include virtual processors from the first set of virtual processors.

20. (original) The computer program of claim 11, wherein executing at least one transaction using the first set of virtual processors simultaneously with executing at least one transaction using the second set of virtual processors comprises executing a first transaction having at least first and second tasks, the first task executed by one or more virtual processors of the first set and the second task executed by one or more virtual processors of the second set.

21. (currently amended) A massively parallel processing system in which database transactions are executed using virtual processors, the processing system comprising:

one or more nodes;

a plurality of CPUs, each of the one or more nodes providing access to one or more CPUs;

a plurality of virtual processors each of the one or more CPUs providing access to one or more virtual processors;

each process configured to manage data stored in one of a plurality of data-storage facilities;

a database-management component configured to execute the transactions by

interconnecting the plurality of nodes;

mapping a first set of the virtual processors across a first subset of the nodes to create a first map with at least one virtual processor being mapped to each node in the first subset;

mapping a second set of the virtual processors across a second subset of the nodes to create a second map with at least one virtual processor being mapped to each node in the second subset;

storing the first map as a first configuration and the second map as a second configuration; and

executing at least one transaction using the first set of virtual processors and simultaneously executing at least one transaction using the second set of virtual processors;

wherein executing a transaction comprises:

identifying tasks necessary to complete the transaction;

identifying storage resources necessary to complete each of the

tasks;

defining one or more groups of virtual processors having access to

the identified storage resources as one or more transaction

groups;

assigning to each task one of the transaction groups that includes

virtual processors having access to resources necessary to

complete the task; and

completing each task using the virtual processors of the assigned

transaction group.

22. (original) The processing system of claim 21, wherein the first subset of the nodes includes all of the one or more nodes and the second subset of nodes includes all of the one or more nodes except for at least one failed node.

23. (original) The processing system of claim 22, wherein the database-management component is also configured to execute the transactions by:

mapping a third set of virtual processors across a third subset of the nodes

to create a third map with at least one virtual processor being

mapped to each node in the third subset;

storing the third map as a third configuration; and wherein the third subset

of the nodes includes all of the one or more nodes.

24. (original) The processing system of claim 23, wherein the third subset includes at least one restored node.

25. (original) The processing system of claim 22, wherein the at least one transaction using the first set of virtual processors does not use any virtual processors mapped to the at least one failed node.

26. (original) The processing system of claim 22, wherein the database-management component is also configured to execute the transactions by:

halting an impacted transaction, the impacted transaction using the first set of virtual processors including at least one virtual processor mapped to one of the at least one failed nodes.

27. (original) The processing system of claim 21, wherein the first and second configurations are stored as a configuration table and configurations are removed from the table when they are no longer associated with an executing transaction.

28. (cancelled)

29. (currently amended) The processing system of claim 21 claim 28, wherein using the first set of virtual processors consists of completing tasks assigned to transaction groups that include virtual processors from the first set of virtual processors.

30. (original) The processing system of claim 21, wherein executing at least one transaction using the first set of virtual processors simultaneously with executing at least one transaction using the second set of virtual processors comprises executing a first transaction having at least first and second tasks, the first task executed by one or more virtual processors of the first set and the second task executed by one or more virtual processors of the second set.